

Альперин М.И., Самсонова Э.Р., Решетников Р.С.

Alperin M.I., Samsonova E.R., Reshetnikov R.S.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ ОБУЧЕНИЯ В МАЛЫХ ГРУППАХ

SOFTWARE FOR TRAINING IN SMALL GROUPS

emiliya_samsonova@mail.ru

ФГАОУ ВПО «УрФУ имени первого Президента России Б.Н.Ельцина»

г. Екатеринбург



В данной статье рассматривается метод группового обучения, который представляет собой одну из наиболее актуальных и перспективных форм организации учебного процесса. Авторы подтверждают свои теоретические выводы экспериментом, проведенным со студентами старших курсов специальности «Информационные системы и технологии». Кроме того, статья включает описание возможностей программы, адаптированной для выполнения общего задания в группах.

This article focuses on the method of group training which represents one of the most actual and perspective forms of the organization of educational process. Authors confirm the theoretical conclusions with the experiment made with students of specialty "Information Systems and Technologies". Besides, article includes the description opportunities of the program adapted for work in small groups.

Внедрение инновационных технологий во многие сферы человеческой деятельности набирает большие обороты. Образование не является исключением. Стремительный скачок в развитии аппаратных средств за последние годы сделал компьютеры достаточно доступными. Поэтому внедрение компьютерных технологий в образование является логичным и необходимым шагом в развитии современного информационного мира в целом. Компьютеризация обучения влечет за собой ряд последствий, которые заставляют пересмотреть классический опыт преподавания. Одной из новых форм организации процесса обучения, вызванной информатизированным обществом, является групповая работа.

Для мировой педагогики нового столетия характерен переход к таким моделям обучения, которые ставят ученика в активную позицию. Учебный процесс строится как поисковая, исследовательская деятельность, в ходе которой происходит обмен мнениями, разворачиваются дискуссии. Наиболее актуальной и перспективной формой организации такого учебного процесса является групповое обучение. Групповая форма обучения предполагает такую организацию деятельности, при которой учащиеся делятся на небольшие рабочие группы для выполнения общего задания.

Данная статья основана на эксперименте, который был проведен со студентами. Все учащиеся были разделены на группы по несколько человек. Каждая группа должна была выполнить задание по созданию программного продукта с учетом всех этапов его жизненного цикла. В результате каждая группа представляла собой отдельную команду, в которой между участниками распределялись роли. По завершении проекта были сделаны интересные наблюдения и дана оценка результативности выполнения работы в группах. Участники сами оценивают возможности друг друга, прикидывают варианты взаимодействия и распределения позиций в группе. Кроме того, коллективное мышление предполагает использование коммуникации между участниками, тесное сотрудничество и обсуждение, так как каждый из них является неотъемлемым звеном в общем процессе работы. Это заставляет каждого члена команды проявить себя, высказать свою точку зрения, узнать мнение других. Достоинствами такой формы

обучения является распределение обязанностей, развитие чувства ответственности за результат совместной деятельности, стимул творческого соревнования.

Работа над проектом проводилась с использованием водопадной модели ведения проекта, которая предварительно была продемонстрирована на лекции. Суть работы в малых группах заключается в следующем. Проект разбивается на отдельные модули, которые распределяются между группами с учетом их возможностей.

Основные этапы разработки

- сбор требований;
- проектирование;
- конструирование (также «реализация» либо «кодирование»);
- интеграция;
- тестирование и отладка (также «верификация»).

Следуя каскадной модели, разработчик переходит от одной стадии к другой строго последовательно. Сначала полностью завершается этап «Сбор требований», в результате чего получается список требований к ПО. После того, как требования полностью определены, происходит переход к проектированию, в ходе которого создаются документы, подробно описывающие для программистов способ и план реализации указанных требований. После того, как проектирование полностью выполнено, программистами выполняется реализация полученного проекта. На следующей стадии процесса происходит интеграция отдельных компонентов, разрабатываемых различными командами программистов. После завершения реализации и интеграции производится тестирование и отладка продукта; на этой стадии устраняются все недочёты, появившиеся на предыдущих стадиях разработки. После этого программный продукт внедряется и обеспечивается его поддержка – внесение новой функциональности и устранение ошибок. Тем самым, каскадная модель подразумевает, что переход от одной фазы разработки к другой происходит только после полного и успешного завершения предыдущей фазы, и что переходов назад либо вперёд или перекрытия фаз – не происходит. Так же в конце каждого этапа проводится инспектирование с целью улучшения качества работы групп. Инспектирующих каждая группа выбирает самостоятельно. В состав инспекционного собрания обязательно должны входить инспектирующий и исполнитель, остальные роли могут быть распределены как на этих двоих, так и назначены другим.

В учебном процессе выбор проекта производится с учетом общего уровня подготовки студентов и их заинтересованности в обучении.

Среди студентов были реализованы несколько проектов, приведем некоторые из них.

1. Ридер текстовых файлов. Студентам был предложен проект программы-ридера, который был разбит на такие модули, как интерфейс

программы, работа с базой данных, смена кодировок, открытие и загрузка файла, поиск файлов на компьютере, форматирование текста.

Каждая группа вела активную работу, и к концу заданного срока проект был успешно выполнен.

2. Сильной группе был предложен более крупный проект ридера с возможностью открытия всех популярных форматов электронных книг, таких как: .txt, .html, .fb2, .pdf, .djvu, .epub – возможность вставки комментариев на странице книги, добавления закладок, поиска информации по книгам в интернете и ведение базы данных по книгам с хранением подробных данных по каждой из них. Студенты, помимо общения внутри группы, вели активное обсуждение между собой, также работа над проектом велась не только на занятиях, но и самостоятельно в свободное время. В результате, к концу срока были завершены два цикла спиральной модели ведения проекта.

3. Слабая группа реализовывала набор мини-игр. Каждая малая группа выполняла отдельную игру. В данной группе активность была меньше, но к концу срока все было завершено.

4. Небольшой группе студентов был предложен проект списка воспроизведения для музыкального проигрывателя с возможностью поиска файлов, загрузки и сохранения списков, сортировка списка, поиска в нем по ключевым словам, чтение и изменение тегов разных версий. Модуль воспроизведения музыки был предоставлен группам руководителем проекта. Разным группам выдавались задания разной сложности в зависимости от их возможностей, и к концу срока проект также был успешно завершен.

В данной статье рассматривается программа, предоставляющая возможность организации работы в группе. Для пользователей программы создаются необходимые условия для решения задачи в группе, а именно возможность обмениваться результатами проделанной работы с другими участниками команды, обсуждения текущих вопросов, отслеживание состояния выполнения работы.

Преподаватель делит учащихся на небольшие группы и выдает общее задание на каждую группу. Разбиение на команды происходит по усмотрению преподавателя, но предпочтительным вариантом является объединение учащихся по уровню знаний и способностей так, чтобы команды были одинаковыми. Члены каждой группы распределяют роли между собой, дробят общую задачу на подзадачи, назначают ответственного за каждую подзадачу. В каждой команде ведется отчетность о проделанной работе, представленная в виде общей и локальной документации. Участник команды, которому отведена роль регистратора, занимается документированием, что включает в себя задачи по сбору C и D требований на всех стадиях проекта. Эта общая документация предоставляется преподавателю, в то время как локальная ведется внутри команды.

Доступ к программе возможен только для зарегистрированных пользователей. Система различает два типа пользователей: участник команды (студент) и преподаватель. Для последнего разрешены все действия на чтение и редактирование документов, а также другие функции, связанные

с администрированием системы. Участнику команды доступен весь функционал, необходимый для выполнения только своей работы и просмотра выполненного задания другим членом команды.

При входе в систему пользователь попадает на свою индивидуальную страницу, на которой отображается его личная информация и список подзадач. Он сам определяет подзадачи, а преподаватель устанавливает дату ее выполнения. Исходя из качества и скорости выполнения поставленных подзадач формируется рейтинг студентов. Такая возможность позволяет преподавателю объективно оценивать работу каждого участника группы.

Одним из преимуществ выбора этой программы для организации работы в группах является общее хранилище документации. Каждый участник команды имеет возможность просматривать все документы, расположенные на сервере, а также он может добавлять свои документы и редактировать существующие. Такая организация хранения документов удобна тем, что все файлы, необходимые для выполнения задания, находятся в одном месте и любой участник группы может просмотреть их в удобное для него время. Каждый член команды имеет возможность редактирования хранящихся на сервере документов, при этом все изменения сохраняются в отдельный файл, имя которого формируется из фамилии редактора и даты.

Данная программа предоставляет возможность совместного редактирования кода. Благодаря этому, участники группы могут дистанционно выполнять задание в режиме on-line. Каждый член команды следит за действиями другого, и все изменения, которые вносят другие участники, мгновенно отображаются на экранах у всей группы.

Контроль за состоянием выполнения работы каждого участника команды – неотъемлемый модуль при разработке в групповых проектах. Каждый участник команды дробит поставленную перед ним задачу на подзадачи и вносит их в свой список подзадач. Исполнитель имеет право оставлять комментарии под каждой из подзадач, а также информировать о готовности ее выполнения. Только преподаватель может закрыть задачу.

Программа находится в разработке.

Обучение в малых группах показывает свою эффективность, повышение интенсивности проведения практических занятий по сравнению с традиционными методами. Благодаря организации такой работы повышается общение между студентами, что способствует повышению количества и качества усваиваемого материала. Даже слабые студенты из-за образования малых групп получают мотивацию к обучению и усваивают материал значительно лучше.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Брауде Эрик Дж. Технология разработки программного обеспечения, Санкт-Петербург, 2004. – 656 с.
2. Выготский Л. С. Психология развития человека, Москва, 2005. – 1136 с.